IN THE SPECIFICATION

Please amend the paragraph beginning at page 1, line 4, as follows;

This application claims priority to and all the benefits of United States Provisional Patent Application Serial No. 60/241,233, filed on October 18, 2000 and entitled "Distributed Multiprocessing System".

Please amend the paragraph beginning at page 2, line 22, as follows;

The subject invention overcomes the deficiencies in the prior art by providing a distributed multiprocessing system comprising a first node and a second node with the nodes being separated from each other. A first processor is disposed within the first node for processing information ~~at a first station~~ and for assigning a first address to a first processed information. A first real memory location is disposed within the first node for storing processed information at the first node. A second processor is disposed within the second node and processes information ~~at a second station~~ and assigns a second address to a second processed information. A second real memory location is disposed within the second node for storing processed information at the second node. A central signal routing hub is interconnected between the first and second processors. An indexer is connected to the routing hub for indexing the first and second nodes to define different destination addresses for each of the nodes. ~~Specifically, a~~ A first communication link interconnects the first ~~processor~~ node and the hub for transmitting the first processed information between the first processor of the first node and the hub without storing the processed information within the first real memory location of the first node. A second communication link interconnects the

second ~~processor~~ node and the hub for transmitting the second processed information between the second processor of the second node and the hub without storing the processed information within the second real memory location of the second node. The central routing hub includes a sorter for receiving at least one of the first and second processed information from at least one of the first and second nodes ~~processors~~, thereby defining at least one sending node ~~processor~~. The hub and sorter also ~~identify~~ associate a destination of at least one of the first and second addresses of the first and second processed information, respectively, with at least one of the destination addresses. Finally, the hub and sorter send at least one of the first and second processed information without modification from the hub over at least one of the communication links to at least one of the first and second nodes ~~processors~~ associated with the destination address, thereby defining at least one addressed node ~~processor~~. The first and second real memory locations store processed information received from the hub.

Please amend the paragraph beginning at page 3, line 15, as follows;

The subject invention also includes a method of communicating across the distributed multiprocessing system having the first node with the first processor and the first real memory location. The system also has the second node with [[and]] the second processor and the second real memory location. The method comprising the steps of; indexing the first and second nodes to define different destination addresses for each of the nodes; processing information within ~~at least one of~~ the first processor of the first node ~~and second processors~~; addressing the processed information using at least one of the destination

addresses; transmitting the processed information from ~~at least one of~~ the first processor of the first node ~~and second processors~~ across ~~at least one of~~ the first communication link ~~links~~ toward the hub without storing the processed information in the first real memory location of the first node, thereby defining ~~at least one~~ a sending node ~~processor~~; receiving the processed information along with the destination address within the hub; identifying the destination ~~of the~~ address for the transmitted processed information within the hub; [[and]] sending the processed information without modification from the hub over at least one of the communication links to at least one of the first and second nodes ~~processors~~ associated with the destination address, thereby defining at least one addressed node ~~processor~~; and storing the processed information within the real memory location of the addressed node.

Please amend the paragraph beginning at page 4, line 2, as follows;

In addition, the unique configuration of the subject invention may be practiced without the hub. In particular, first and second real memory locations are connected to the first and second processors within the first and second nodes~~, respectfully,~~ for storing received processed information. An indexer is provided for indexing ~~said~~ the first and second nodes ~~processors~~ to define a different identifier ~~code~~ for each of ~~said processors~~ the nodes for differentiating ~~said processors~~ the nodes. Further, ~~said~~ the first and second nodes ~~processors~~ each include virtual memory maps of each identifier ~~code~~ such that said first and second processors can address and forward processed information to each of ~~said~~ the indexed nodes ~~processors~~ within ~~said~~ the system.

Please amend the paragraph beginning at page 4, line 10, as follows;

The subject invention eliminating the hub also includes the steps of indexing the first and second nodes ~~processors~~ to define a different identifier ~~code~~ for each of the nodes ~~processors~~ for differentiating the nodes ~~processors~~; creating a virtual memory map of each of the identifiers ~~codes~~ within each of the first and second nodes ~~processors~~ such that the first and second processors can address and forward processed information to each of the indexed nodes ~~processors~~ within the system; and storing the processed information within the real memory location of the addressed node ~~processor~~.

Please amend the paragraph beginning at page 6, line 18, as follows;

Referring to the Figures, wherein like numerals indicate like or corresponding parts throughout the several views, a distributed multiprocessing system is generally shown at 30 in Figure 1. The system 30 comprises a plurality of modules or nodes 1-6 interconnected by a central signal routing hub 32 to preferably create a star topology configuration. As illustrated, there are six nodes 1-6 connected to the hub 32 with each of the nodes 1-6 being indexed with a particular code or identifier. As an example of an identifier ~~a code~~, numerical indicators 1 through 6 are illustrated. As appreciated, any suitable alpha/numeric indicator may be used to differentiate one node from another. The shape, configuration, and orientation of the hub 32, which is shown as an octagon shape, is purely illustrative and may be altered to meet any desired need.

Please amend the paragraph beginning at page 8, line 13, as follows;

The processors may be of different sizes and speeds. ~~For example, node 6 may have a 1,500 MFbps processor and the remaining nodes may have a 300 MFbps processors.~~ The size and speed of the processor may be varied to satisfy a multitude of design criteria. Typically, the processor will only be of a size and speed to support the tasks or operation which are associated with the node 1-6. Further, the processors can be of different types which recognize different computer formats and languages.

Please amend the paragraph beginning at page 8, line 19, as follows;

Nodes 1 and 2 will now be discussed in greater detail. The first node, node 1, includes a first processor 40 and the second node, node 2, includes a second processor 42. The first 40 and second 42 processors are indexed in concert with nodes 1 and 2 to define a different <u>identifier</u> ~~code~~ for each of the processors 40, 42 for differentiating the processors 40, 42 in the same fashion as the nodes 1-6 are differentiated. In particular, an indexer 73, which is discussed in greater detail below, is included for indexing the first 40 and second 42 processors to define the different <u>identifier</u> ~~code~~ for each of the processors 40, 42 for differentiating the processors 40, 42 and the nodes 1-6.

Please amend the paragraph beginning at page 9, line 14, as follows;

As shown in Figures 5 and 7, the first 40 and second 42 processors further include a hardware portion 48 for assigning the first and second addresses to the first and second processed information, respectively. In particular, the hardware portion 48 assigns a

destination address onto the processed information corresponding to the identifier ~~indicative of the code~~ of an addressed node 1, 2 ~~processor~~. The hardware portion 48 also conforms or rearranges the data or information to an appropriate format. As discussed above, the processors 40, 42 can be of different types which recognize different computer formats. Hence, the hardware portion 48 ensures that the proper format is sent to the addressed node 1, 2 ~~processor~~. However, the addresses are preferably of a common format such that the hub 32 commonly recognizes these signals. Examples of the processors 40, 42 operation are discussed below in greater detail.

Please amend the paragraph beginning at page 9, line 25, as follows;

A first memory space 50 is connected to the first processor 40 and a second memory space 52 is connected to the second processor 42. As shown in Figures 4 and 6, the first 50 and second 52 memory spaces are shown in greater detail, respectively. A first real memory location 54 is disposed within the first memory space 50 and is connected to the hardware portion 48 of the first processor 40. Similarly, a second real memory location 56 is disposed within the second memory space 52 and is connected to the hardware portion 48 of the second processor 42. During operation, the hardware portion 48 assigns a memory address onto the processed information corresponding to ~~indicative of~~ the memory location of an addressed node 1, 2 ~~processor~~. The first 54 and second 56 real memory locations can therefore store received processed information, which is also discussed in greater detail below. The first [[54]] 40 and second 42 processors ~~56 real memory locations~~ are not capable of reading the real memory of another processor. In other words, the processors of a

particular node 1-6 can read its own real memory within its own real memory locations but cannot read the real memory stored within a real memory location of another processor.

Please amend the paragraph beginning at page 10, line 19, as follows;

Also illustrated within the first 50 and second 52 memory spaces at Figures 4 and 6, are first 58 and second 60 virtual memory maps. The first 50 [[40]] and second 52 memory spaces ~~42 processors~~ each include virtual memory maps 58, 60 of each identifier ~~code disposed within each of the first 40 and second 42 processors~~ for each node 1-6 such that the first 40 and second 42 processors can address and forward processed information to each of the indexed nodes 1-6 ~~processors~~ within the system 30. The virtual memory maps 58, 60 are essentially a means for the processors 40, 42 to be able to address each other processor or node 1-6 within the system 30. The operation and specifics of the virtual memory maps 58, 60 will be discussed in greater detail below.

Please amend the paragraph beginning at page 11, line 16, as follows;

As also shown in Figures 5 and 7, each task 62 includes at least a pair of pointers 64, 66 for directing a flow of data from a sending node 1, 2 ~~processor~~ to a destination node 1, 2 ~~processor~~. The pointers 64, 66 are illustrated as branching off of the fourth task 62 in Figure 5 and the third task 62 in Figure 7. As should be appreciated, there are pointers 64, 66 associated with each of the tasks 62 such that there is a continuous stream of information. ~~Each pair of~~ The pointers 64, 66 includes a next task pointer 64 for directing the sending node 1, 2 ~~processor~~ to a subsequent task 62 to be performed, and at least one data

destination pointer 66 for directing the sending node 1, 2 to forward ~~sending~~ the processed information to the hub 32. Preferably, there is only one next task pointer 64 such that there is a clear order of operation for the processors 40, 42. Conversely, there may be any number of data destination pointers 66 such that the sending node 1, 2 ~~processor~~ may simultaneously forward processed information to a multitude of addressed nodes 1-6 ~~processors~~. Further, each of the processed information sent to the multitude of addressed nodes 1-6 ~~processors~~ may be different.

Please amend the paragraph beginning at page 12, line 24, as follows;

As discussed above, an indexer 73 is provided for indexing or organizing the first 40 and second 42 processors to define the different identifiers ~~codes~~ for each of the processors 40, 42, which differentiates the processors 40, 42 and the nodes 1-6. Preferably, the indexer 73 is disposed within the hub 32. Hence, when the nodes 1-6 are initially connected to the hub 32, the indexer 73 within the hub 32 begins to organize the nodes 1-6 in a particular order. This is how the entire organization of the system 30 begins. The hub 32 and indexer 73 also create the mapping within the memory spaces 50, 52 ~~processors 40, 42~~ as part of this organization. As discussed above the mapping includes the first 58 and second 60 virtual memory maps of the nodes 1, 2 ~~first 40 and second 42 processors~~. The virtual memory maps 58, 60 outline each identifier ~~code disposed within each of the processors~~ for each node 1-6 such that the processors can address and forward processed information to each of the indexed nodes 1-6 ~~processors~~ within the system 30.

Please amend the paragraph beginning at page 13, line 12, as follows;

As shown in Figure 3, the central routing hub 32 includes a sorter 72 for receiving at least one of the first and second processed information from at least one of the first 40 and second 42 processors. By receiving the processed information, at least one sending node 1-6 ~~processor~~ is defined. Each of the first 40 and second 42 processors of the nodes 1, 2 may send processed information or only one of the first 40 and second 42 processors of the nodes 1, 2 may send processed information. In any event, at least one of the nodes 1-6 ~~first 40 and second 42 processors~~ will be deemed as a sending node 1-6 ~~processor~~.

Please amend the paragraph beginning at page 13, line 19, as follows;

The hub 32 and sorter 72 also identify a destination of at least one of the first and second addresses of the first and second processed information, respectively to define the destination address. Finally, the hub 32 and sorter 72 send at least one of the first and second processed information without modification from the hub over at least one of the communication links 68, 70 to at least one of the nodes 1, 2 ~~first 40 and second 42 processors~~. The node 1, 2 ~~processor~~ to which the information is being sent defines at least one addressed node 1, 2 ~~processor~~. The sorter 72 includes hardware 74 for determining the destination addresses of the addressed nodes 1-6 ~~processors~~.

Please amend the paragraph beginning at page 14, line 23 as follows;

The distributed multiprocessing system 30 can include any number of additional features for assisting in the uninterrupted flow of data through the system 30. For example,

a counter may be included to determine, [[and]] control, and limit a number of times processed information is sent from a sending node to an addressed node 1-6 ~~processor~~. A sequencer may also be included to monitor and control a testing operation as performed by the system 30. In particular, the sequencer may be used to start the testing, perform the test, react appropriately to limits and events, establish that the test is complete, and switch off the test.

Please amend the paragraph beginning at page 15, line 6, as follows;

Referring to Figure 8, an alternative embodiment of the system 30 is shown wherein there are only two nodes 1, 2 and the hub 32 is eliminated. In this embodiment, a single communication link 68 interconnects the first processor 40 with the second processor 42 for transmitting the first and second processed information between the first 40 and second 42 processors of the nodes 1, 2. An indexer (not shown in this Figure) indexes the first 40 and second 42 processors to define a different identifier ~~code~~ for each of the processors 40, 42 and nodes 1, 2 in a similar ~~similarly~~ manner as above. The first 50 [[40]] and second 52 memory spaces ~~42 processors~~ also each include virtual memory maps of each identifier ~~code~~ such that the nodes 1, 2 ~~first 40 and second 42 processors~~ can address and forward processed information to each other. There are also first 54 [[50]] and second 56 [[52]] real memory locations for storing received processed information. The unique architecture allows the two nods 1, 2 to communicate in a virtually seamless manner.

Please amend the paragraph beginning at page 15, line 17, as follows;

Specifically, the method of communicating between the nodes 1, 2 and the first 40 and second 42 processors includes the steps of initially indexing the nodes 1, 2 ~~first 40 and second 42 processors~~ to differentiate the nodes 1, 2 ~~processors 40, 42~~. Then the virtual memory maps of each of the identifiers ~~codes~~ is created within each of the first 50 [[40]] and second 52 memory spaces ~~42 processors~~ such that the first 40 and second 42 processors can address and forward processed information to each other. The processed information is transmitted by utilizing the virtual memory map of the sending node 1, 2 ~~processor~~, which may be from either node 1, 2, from the sending node 1, 2 ~~processor~~ across the communication link toward the addressed node 1, 2 ~~processor~~, which is the corresponding opposite node 1, 2. The processed information is then received along with the address in the addressed node 1, 2 ~~processor~~ and the processed information is stored within the real memory location of the addressed node 1, 2 ~~processor~~.

Please amend the paragraph beginning at page 16, line 15, as follows;

Specifically, the indexer first indexes the first 32 and second 84 hubs to define a master hub 32 and secondary hub 84. In the illustrated example, hub number 1 is the master hub 32 and hub number 2 is the secondary hub 84. A key 88 is disposed within one of the first 32 and second 84 hubs to determine which of the hubs 32, 84 will be defined as the master hub. As illustrated, the key 88 is within the first hub 32. The indexer also indexes the nodes 1-8 and processors to redefine the identifiers ~~codes~~ for each of the nodes 1-8 for differentiating the processors and nodes 1-8. When the first or master hub 32 is connected

to the second or secondary hub 84 the entire virtual memory maps of each <u>node 1-8</u> ~~processor~~ connected to the first hub 32 is effectively inserted into the virtual memory maps of each <u>node 1-8</u> ~~processor~~ connected to the second hub 84 and vise versa. Hence, each hub 32, 84 can write to all of the nodes 1-8 in the new combined or cascaded system 30 as shown in Figure 9.

Please amend the paragraph beginning at page 18, line 7, as follows;

To maintain the continuous flow of information, the system 30 further includes the step of directing the sending <u>node 1-6</u> ~~processor~~, which in this example is the first processor 40 of node 1, to a subsequent task 62 to be performed within the first processor 40 while simultaneously sending the processed information across one of the communication links 68, 70 to the hub 32. This step is accomplished by the use of the tasks 62 and pointers 64, 66. As shown, the first task 62 is first completed and then the first processor 40 proceeds to the second task 62. The pointers 64, 66 within the first task 62 direct the flow of the first processor 40 to the second task 62. Specifically, the data destination pointer 66 is silent and the next task pointer 64 indicates that the second task 62 should be the next task to be completed. The second task 62 is then completed and the first processor 40 proceeds to the fourth task 62. In this step, the next task pointer 64 of the second task 62 indicates to the first processor 40 that the fourth task 62 should be next, thereby skipping over the third task 62. The fourth task 62 is completed and the next task pointer 64 directs the flow to another task 62. The data destination pointer 66 of the fourth task 62 indicates that the information as processed after the fourth task 62 should be sent to the hub 32. The flow of information

from the first task 62 to the second task 62 to the fourth ~~forth~~ task 62 is purely illustrative and is in now way intended to limit the subject application.

Please amend the paragraph beginning at page 18, line 24, as follows;

The processed information from the fourth task 62 is then addressed and transmitted from the first processor 40 across at least one of the communication links 68, 70 toward the hub 32. As discussed above, the communication links 68, 70 are preferably unidirectional. Hence, the step of transmitting the processed information is further defined as transmitting the processed information across the first incoming transmission line 76 in only one direction from the first processor 40 to the hub 32 to define a send-only system 30. The transmitting of the processed information is also further defined by transmitting the data along with executable code from the sending node 1-6 ~~processor~~ to the addressed node 1-6 ~~processor~~. As appreciated, the first 40 and second 42 processors initially do not have any processing capabilities. Hence, the executable code for the processors 40, 42 is preferably sent to the processors 40, 42 over the same system 30. Typically, the executable code will include a command to instruct the processors 40, 42 to process the forwarded data in a certain fashion. It should also be noted that the transmitting of the processed information may be a command to rearrange or reorganize the pointers of the ~~addressed~~ processor of the addressed node 1-6. This in turn may change the order of the tasks which changes the processing of ~~the addressed~~ this processor. As appreciated, the transmitted processed data may include any combination of all or other like features.

Please amend the paragraph beginning at page 19, line 17, as follows;

The processed information is preferably addressed by the data destination pointer 66 directing the flow to the first virtual memory map 58 of node 1 and pointing to a destination node. The step of addressing the processed information is further defined as assigning a destination address onto the processed information corresponding to an identifier ~~indicative of a code~~ of an addressed node 1-6 ~~processor~~. The step of addressing the processed information is further defined as assigning a memory address onto the processed information corresponding to ~~indicative of~~ the memory location of the addressed node 1-6 ~~processor~~, i.e., node 2. In this example the destination node, destination address, and memory address will be node 2 while the originating node will be node 1.

Please amend the paragraph beginning at page 20, line 2, as follows;

The virtual memory map 58, 60 of each of the identifiers ~~codes~~ is created within each of the nodes 1, 2 ~~first 40 and second 42 processors~~ such that the first 40 and second 42 processors can address and forward processed information to each of the indexed nodes 1, 2 ~~processors~~ within the system 30. As discussed above, the virtual memory map 58, 60 is a means to which the processor can recognize and address each of the other processors in the system 30. By activating the data destination pointer 66 to direct a sending node 1, 2 to send information to the hub 32, node 1 is then defined as [[a]] the sending node 1 ~~processor~~. As shown in Figure 16, the data destination pointer 66 directs the processed information to node 2 in the first virtual memory map 58 such that the destination address of node 2 will be assigned to this information.

Please amend the paragraph beginning at page 20, line 14, as follows;

Referring to Figure 18, the destination of the address for the transmitted processed information is identified within the hub 32 and the processed information is sent without modification over the second communication link 70 to, in this example, the second processor 42 of node 2. The step of sending the processed information without modification is further defined as sending the processed information over the second outgoing transmission line 82 in only one direction from the hub 32 to the second processor 42 to further define the send-only system 30. In this example, the hub 32 determines that the destination of the address is for node 2 which defines node 2 as an addressed node ~~processor~~ with the destination address.

Please amend the paragraph beginning at page 20, line 23, as follows;

As shown in Figure 19, the processed information is then stored within the second real memory location 56 of the addressed node 2 ~~second processor 42~~ wherein the second processor 42 can utilize the information as needed. The processed information may be stored within the categorized message areas or locations of the second real memory location 56 in accordance with the associated memory address. ~~To save on memory space, the~~ The destination address (of node 2) may be stripped from sent processed information before the information is stored in the second real memory location 56.

Please amend the paragraph beginning at page 21, line 22, as follows;

Another example of communicating across the subject system 30 is illustrated in Figure 20 wherein node 2 communicates with itself. The information is processed within the second processor 42 of node 2 by proceeding through a number of tasks 62. The processed information is then addressed and transmitted from the second processor 42 across the second incoming transmission line 80 toward the hub 32. The processed information is addressed by the data destination pointer 66 directing the flow to the second virtual memory map 60 and pointing to the destination node. A destination address and a memory address are then assigned to the information. In this example the destination node, destination address, and memory address will be node 2 while the originating node will also be node 2. By activating the data destination pointer 66 to direct a sending node 1-6 to send information to the hub 32, node 2 is defined as [[a]] the sending node 2 ~~processor~~. The processed information, along with the address, is then received within the hub 32. The destination of the address for the transmitted processed information is identified within the hub 32 and the processed information is sent without modification from the hub over the second outgoing transmission line 82 to the designated node 1-6 ~~processor~~. In this example, the hub 32 determines that the destination of the address is for node 2 which defines node 2 as an addressed node 2 ~~processor~~ with the destination address. The processed information is sent across the second outgoing transmission line 82 back to the second processor 42 within node 2. The processed information is then stored within the second real memory location 56 of the addressed ~~second processor 42 of~~ node 2. Node 2 has now successfully written information to itself.